Available online at www.sciencedirect.com

ScienceDirect

Chinese Journal of Aeronautics 20(2007) 346-352

Chinese Journal of Aeronautics

www.elsevier.com/locate/cja

The Marginal Rao-Blackwellized Particle Filter for Mixed Linear/Nonlinear State Space Models

Yin Jianjun[a], Zhang Jianqiu[a,*], Mike Klaas[b]

[a]*Department of Electronic Engineering, Fudan University, Shanghai 200433, China*

[b]*Department of Computer Science, University of British Columbia, Vancouver V6T1Z4, Canada*

Received 7 July 2006; accepted 20 March 2007

Abstract

In this paper, the marginal Rao-Blackwellized particle filter (MRBPF), which fuses the Rao-Blackwellized particle filter (RBPF) algorithm and the marginal particle filter (MPF) algorithm, is presented. The state space is divided into linear and non-linear parts, which can be estimated separately by the MPF and the optional Kalman filter. Through simulation in the terrain aided navigation (TAN) domain, it is demonstrated that, compared with the RBPF, the root mean square errors (RMSE) and the error variance of the nonlinear state estimations by the proposed MRBPF are respectively reduced by 29% and 96%, while the unique particle count is increased by 80%. It is also found that the MRBPF has better convergence properties, and analysis has shown that the existing RBPF is nothing more than a special case of the MRBPF.

Keywords: signal processing; marginal Rao-Blackwellized particle filter; simulation; mixed linear/nonlinear; terrain aided navigation

1 Introduction

Let the following general model of a nonlinear state space system[1] be considered:

$$\boldsymbol{x}_{t+1} = f(\boldsymbol{x}_t, \boldsymbol{\omega}_t) \quad (1.1)$$

$$\boldsymbol{y}_t = h(\boldsymbol{x}_t, \boldsymbol{e}_t) \quad (1.2)$$

where $\boldsymbol{\omega}_t$ and $\boldsymbol{e}_t$ denote the known independent process and measurement noise respectively, $\{\boldsymbol{x}_t\}$ is the state of the system complying with the Markov process with $\boldsymbol{x}_t \in \mathbf{R}^{n_x}$ (n_x denotes the dimension of the state), $\{\boldsymbol{y}_t\}$ is the measurements of the system with $\boldsymbol{y}_t \in \mathbf{R}^{n_y}$ (n_y denotes the dimension of the measurement), and both f and h are known nonlinear functions. Again, let $\boldsymbol{x}_{0:t} = \{\boldsymbol{x}_0, \boldsymbol{x}_1, \cdots, \boldsymbol{x}_t\}$, $\boldsymbol{y}_{0:t} = \{\boldsymbol{y}_0, \boldsymbol{y}_1, \cdots, \boldsymbol{y}_t\}$, and $p(\boldsymbol{x}_0 \mid \boldsymbol{x}_{-1}) = p(\boldsymbol{x}_0)$. Here, the purpose is supposed to estimate the posterior probability density function (PDF) $p(\boldsymbol{x}_t \mid \boldsymbol{y}_{0:t})$ by the following equations:

$$\left.\begin{aligned} p(\boldsymbol{x}_t \mid \boldsymbol{y}_{0:t-1}) &= \int p(\boldsymbol{x}_t \mid \boldsymbol{x}_{t-1})\, p(\boldsymbol{x}_{t-1} \mid \boldsymbol{y}_{0:t-1}) \mathrm{d}\boldsymbol{x}_{t-1} \\ p(\boldsymbol{x}_t \mid \boldsymbol{y}_{0:t}) &= \frac{p(\boldsymbol{y}_t \mid \boldsymbol{x}_t)\, p(\boldsymbol{x}_t \mid \boldsymbol{y}_{0:t-1})}{\int p(\boldsymbol{y}_t \mid \boldsymbol{x}_t)\, p(\boldsymbol{x}_t \mid \boldsymbol{y}_{0:t-1}) \mathrm{d}\boldsymbol{x}_t} \end{aligned}\right\} \quad (2)$$

Unfortunately, it is almost unlikely to obtain a closed-form expression of Eq.(2). One special case appears when Eq.(1.1) and Eq.(1.2) are used to describe a linear Gaussian model, which leads to the famous Kalman filter (KF) recursions. In more common cases, to calculate the integrals in Eq.(2), numerical approximation methods must be used, in which the extended Kalman filter (EKF)[2] is most popular. However, the linearization process of the EKF is liable to large errors threatening the convergence of the algorithm, particularly for models with high nonlinearity.

A recently-popularized technique for numerical approximation, termed as the particle filter (PF)[1,3-6],

*Corresponding author. Tel.: +86-21-55664226.
E-mail address: jqzhang01@fudan.edu.cn
Foundation item: National Natural Science Foundation of China (60572023)

offers a general tool for the state estimation of nonlinear non-Gaussian systems. The core idea behind the PF is to use samples (particles) to approximate the concerned distributions. The classic PF formulae give the estimations of the joint distribution $p(\boldsymbol{x}_{0:t} \mid \boldsymbol{y}_{0:t})$. However, in many applications such as tracking, a marginal distribution $p(\boldsymbol{x}_t \mid \boldsymbol{y}_{0:t})$ is required. When using PF, this is approximated by dropping particles of the states $\boldsymbol{x}_0, \boldsymbol{x}_1, \cdots, \boldsymbol{x}_{t-1}$ from the joint distribution. Obviously, this is an inefficient approach. What is needed is to obtain particles $\boldsymbol{x}_0, \boldsymbol{x}_1, \cdots, \boldsymbol{x}_t$ for the PF in the state space $(\mathbf{R}^{n_x})^{t+1}$, whose dimension is t times higher than $\mathbf{R}^{n_x}$ which is the state space of $\boldsymbol{x}_t$. This results in higher variance of the importance weights[7-8]. In order to deal with this problem, the marginal particle filtering (MPF) algorithm[7], which directly estimates the marginal distribution $p(\boldsymbol{x}_t \mid \boldsymbol{y}_{0:t})$, is proposed. The literature shows a clear superiority of the MPF over the PF in terms of the importance weight variance.

In practices, very often the nonlinear states constitute only one part of the whole state space, for instance, integrated navigation systems[9-11]. In this case, if the PF is used to estimate all parts of the state space, a large number of particles for each state must be acquired, which, owing to being computationally-prohibitive, may restrict the applicability of the PF in real-time applications[3]. Recently, there is presented the Rao-Blackwellized particle filter (RBPF)[3,9], which is sometimes referred to as marginalized PF[10-11]. The RBPF optimizes the PF by marginalizing out the linear states which are estimated with the KF, and the nonlinear states are estimated still with the PF so as to reduce the likelihood of filtering divergence, the particle number, and the computational intensity. However, since the classic PF is still used by the RBPF for nonlinear state estimation, the same problems also remained associated with the PF. In this paper, by introducing the idea of marginal filtering into RBPF, the marginal Rao-Blackwellized particle filter (MRBPF) algorithm is put forward. The analytical results indicate that the existing RBPF is nothing but a special case of the proposed MRBPF. Moreover, the terrain aided navigation (TAN) simulation results show that, compared with the RBPF, while the MRBPF reduces the root mean square errors (RMSE) and error variance of the nonlinear state estimation by about 29% and 96% respectively, and, at the same time, increases the unique particle count by 80%, it achieves better convergence as well.

2 The MRBPF Algorithm

Supposing a system is divided into two parts: a linear and a nonlinear, and the noise is additive, the Eq.(1) can be expressed as follows[3]

$$\boldsymbol{x}_{t+1}^{\mathrm{n}} = f(\boldsymbol{x}_t^{\mathrm{n}}) + \boldsymbol{A}_t^{\mathrm{n}} \boldsymbol{x}_t^{\mathrm{l}} + \boldsymbol{B}_t^{\mathrm{n}} \boldsymbol{\omega}_t^{\mathrm{n}} \tag{3.1}$$

$$\boldsymbol{x}_{t+1}^{\mathrm{l}} = \boldsymbol{A}_t^{\mathrm{l}} \boldsymbol{x}_t^{\mathrm{l}} + \boldsymbol{B}_t^{\mathrm{l}} \boldsymbol{\omega}_t^{\mathrm{l}} \tag{3.2}$$

$$\boldsymbol{y}_t = h(\boldsymbol{x}_t^{\mathrm{n}}) + \boldsymbol{e}_t \tag{3.3}$$

where $\boldsymbol{x}_t^{\mathrm{n}}$ and $\boldsymbol{x}_t^{\mathrm{l}}$ denote the nonlinear and linear states respectively, and $[\boldsymbol{x}_t^{\mathrm{n}}, \boldsymbol{x}_t^{\mathrm{l}}]^{\mathrm{T}} = \boldsymbol{x}_t$. $\boldsymbol{\omega}_t$ is the process noise given by

$$\boldsymbol{\omega}_t = \begin{bmatrix} \boldsymbol{\omega}_t^{\mathrm{n}} \\ \boldsymbol{\omega}_t^{\mathrm{l}} \end{bmatrix} \sim N\left(0, \begin{pmatrix} \boldsymbol{Q}_t^{\mathrm{n}} & \boldsymbol{S}_t \\ \boldsymbol{S}_t^{\mathrm{T}} & \boldsymbol{Q}_t^{\mathrm{l}} \end{pmatrix}\right)$$

where $N(0, \sigma^2)$ denotes the normal distribution with 0 as the mean value and σ^2 the variance. Also, $\boldsymbol{x}_0^{\mathrm{l}} \sim N\left(\hat{\boldsymbol{x}}_{0|-1}^{\mathrm{l}}, \boldsymbol{P}_{0|-1}^{\mathrm{l}}\right)$ and $\boldsymbol{e}_t, \boldsymbol{x}_0^{\mathrm{n}}$ have arbitrary fixed PDFs. Here the target distribution is the posterior distribution $p(\boldsymbol{x}_t \mid \boldsymbol{y}_{0:t})$. By using Bayes' rule,

$$p(\boldsymbol{x}_t \mid \boldsymbol{y}_{0:t}) = p(\boldsymbol{x}_t^{\mathrm{n}}, \boldsymbol{x}_t^{\mathrm{l}} \mid \boldsymbol{y}_{0:t}) = p(\boldsymbol{x}_t^{\mathrm{l}} \mid \boldsymbol{x}_t^{\mathrm{n}}, \boldsymbol{y}_{0:t}) p(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t}) \tag{4}$$

can be achieved.

2.1 Linear state estimation

According to Eq.(3.3), if $\boldsymbol{x}_t^{\mathrm{n}}$ is known, the measurement $\boldsymbol{y}_t$ is conditionally independent, and the first term on the right of Eq.(4) can be simplified as

$$p(\boldsymbol{x}_t^{\mathrm{l}} \mid \boldsymbol{x}_t^{\mathrm{n}}, \boldsymbol{y}_{0:t}) = p\left(\boldsymbol{x}_t^{\mathrm{l}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n}}\right) \tag{5}$$

The process model of Eq.(3) can be rewritten as:

$$\boldsymbol{x}_{t+1}^{\mathrm{l}} = \boldsymbol{A}_t^{\mathrm{l}} \boldsymbol{x}_t^{\mathrm{l}} + \boldsymbol{B}_t^{\mathrm{l}} \boldsymbol{\omega}_t^{\mathrm{l}} \tag{6.1}$$

$$\boldsymbol{z}_t = \boldsymbol{A}_t^{\mathrm{n}} \boldsymbol{x}_t^{\mathrm{l}} + \boldsymbol{B}_t^{\mathrm{n}} \boldsymbol{\omega}_t^{\mathrm{n}} \tag{6.2}$$

where $\boldsymbol{z}_t = \boldsymbol{x}_{t+1}^{\mathrm{n}} - f(\boldsymbol{x}_t^{\mathrm{n}})$. If $\boldsymbol{z}_t$ is regarded as a measurement and $\boldsymbol{x}_t^{\mathrm{l}}$ as the state, then Eq.(6) describes a linear Gaussian model, which enables the states to

be optimized by the KF, and Eq.(5) meets the Gaussian distribution

$$p\left(\boldsymbol{x}_t^{\mathrm{l}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n}}\right)=N\left(\hat{\boldsymbol{x}}_{t\mid t-1}^{\mathrm{l}}, \boldsymbol{P}_{t\mid t-1}^{\mathrm{l}}\right)$$

where $\hat{\boldsymbol{x}}_{t\mid t-1}^{\mathrm{l}}$ and $\boldsymbol{P}_{t\mid t-1}^{\mathrm{l}}$ denote the one-step-ahead predictions of the states and its covariance respectively, which can be obtained by the KF (provided $\boldsymbol{\omega}_t^{\mathrm{n}}$ and $\boldsymbol{\omega}_t^{\mathrm{l}}$ are uncorrelated) through

$$\boldsymbol{k}_t=\boldsymbol{P}_{t\mid t-1}^{\mathrm{l}}(\boldsymbol{A}_t^{\mathrm{n}})^{\mathrm{T}}(\boldsymbol{A}_t^{\mathrm{n}}\boldsymbol{P}_{t\mid t-1}^{\mathrm{l}}(\boldsymbol{A}_t^{\mathrm{n}})^{\mathrm{T}}+\boldsymbol{B}_t^{\mathrm{n}}\boldsymbol{Q}_t^{\mathrm{n}}(\boldsymbol{B}_t^{\mathrm{n}})^{\mathrm{T}})^{-1}$$

$$\hat{\boldsymbol{x}}_{t+1\mid t}^{\mathrm{l}}=\boldsymbol{A}_t^{\mathrm{l}}\left(\hat{\boldsymbol{x}}_{t\mid t-1}^{\mathrm{l}}+\boldsymbol{k}_t\left(\boldsymbol{z}_t-\boldsymbol{A}_t^{\mathrm{n}}\hat{\boldsymbol{x}}_{t\mid t-1}^{\mathrm{l}}\right)\right)$$

$$\boldsymbol{P}_{t+1\mid t}^{\mathrm{l}}=\boldsymbol{A}_t^{\mathrm{l}}(\boldsymbol{P}_{t\mid t-1}^{\mathrm{l}}-\boldsymbol{k}_t\boldsymbol{A}_t^{\mathrm{n}}\boldsymbol{P}_{t\mid t-1}^{\mathrm{l}})(\boldsymbol{A}_t^{\mathrm{l}})^{\mathrm{T}}+\boldsymbol{B}_t^{\mathrm{l}}\boldsymbol{Q}_t^{\mathrm{l}}(\boldsymbol{B}_t^{\mathrm{l}})^{\mathrm{T}}$$

2.2 Nonlinear state estimation

Now comes the estimation of the nonlinear states, i.e. the second part on the right of Eq.(4), $p(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t})$. According to Bayesian inference

$$p(\boldsymbol{x}_t^{\mathrm{n}}, \boldsymbol{y}_{0:t-1})=\int p(\boldsymbol{x}_t^{\mathrm{n}}, \boldsymbol{x}_{t-1}^{\mathrm{n}}, \boldsymbol{y}_{0:t-1})\mathrm{d}\boldsymbol{x}_{t-1}^{\mathrm{n}}=$$
$$\int p(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{x}_{t-1}^{\mathrm{n}}, \boldsymbol{y}_{0:t-1})\,p(\boldsymbol{x}_{t-1}^{\mathrm{n}}, \boldsymbol{y}_{0:t-1})\mathrm{d}\boldsymbol{x}_{t-1}^{\mathrm{n}}$$

then follows

$$p(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t-1})=\int p(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{x}_{t-1}^{\mathrm{n}}, \boldsymbol{y}_{0:t-1})\cdot p(\boldsymbol{x}_{t-1}^{\mathrm{n}} \mid \boldsymbol{y}_{0:t-1})\mathrm{d}\boldsymbol{x}_{t-1}^{\mathrm{n}} \tag{7}$$

and

$$p(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t})=(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n}})p(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t-1})\Big/\int(p(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n}})\cdot$$
$$p(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t-1})\mathrm{d}\boldsymbol{x}_t^{\mathrm{n}}) \propto p(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n}})p(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t-1})= \tag{8}$$
$$p(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n}})\int p(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{x}_{t-1}^{\mathrm{n}}, \boldsymbol{y}_{0:t-1})\,p(\boldsymbol{x}_{t-1}^{\mathrm{n}} \mid \boldsymbol{y}_{0:t-1})\mathrm{d}\boldsymbol{x}_{t-1}^{\mathrm{n}}$$

where

$$p\left(\boldsymbol{x}_{t-1}^{\mathrm{n}} \mid \boldsymbol{y}_{0:t-1}\right)=\sum_{i=1}^{N}w_{t-1}^{(i)}\delta_{\boldsymbol{x}_{t-1}^{\mathrm{n},(i)}}\left(\mathrm{d}\boldsymbol{x}_{t-1}^{\mathrm{n}}\right) \tag{9}$$

where $w_{t-1}^{(i)}$ is particle i normalized weight of time t–1, $\delta(\cdot)$ is Dirac's delta function with the following important property

$$\int_{-\infty}^{+\infty}g(s)\delta(s-t)\mathrm{d}s=g(t)$$

Inserting Eq.(9) in Eq.(7) gives

$$p\left(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t-1}\right)=\sum_{i=1}^{N}w_{t-1}^{(i)}p\left(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{x}_{t-1}^{\mathrm{n},(i)}, \boldsymbol{y}_{0:t-1}\right) \tag{10}$$

Inserting Eq.(10) into Eq.(8) gives

$$p\left(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t}\right)\propto p\left(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n}}\right)\sum_{i=1}^{N}w_{t-1}^{(i)}p\left(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{x}_{t-1}^{\mathrm{n},(i)}, \boldsymbol{y}_{0:t-1}\right) \tag{11}$$

As $\boldsymbol{x}_{0:t-1}^{\mathrm{n}}$ is given and $\boldsymbol{y}_{0:t-1}$ is conditionally independent, then

$$p\left(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{x}_{t-1}^{\mathrm{n}}, \boldsymbol{y}_{0:t-1}\right)=p\left(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n}}\right)$$

thus Eq.(11) can be rewritten as

$$p\left(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t}\right)\propto p\left(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n}}\right)\sum_{i=1}^{N}w_{t-1}^{(i)}p\left(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n},(i)}\right)$$

If the importance density is chosen to be

$$q\left(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t}\right)=\sum_{i=1}^{N}w_{t-1}^{(i)}q\left(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n},(i)}, \boldsymbol{y}_t\right)$$

then the importance weight can be expressed as follows

$$w_t^{(j)}=\frac{p\left(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t}\right)}{q\left(\boldsymbol{x}_t^{\mathrm{n}} \mid \boldsymbol{y}_{0:t}\right)}\propto$$
$$\frac{p\left(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n},(j)}\right)\sum_{i=1}^{N}w_{t-1}^{(i)}p\left(\boldsymbol{x}_t^{\mathrm{n},(j)} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n},(i)}\right)}{\sum_{i=1}^{N}w_{t-1}^{(i)}q\left(\boldsymbol{x}_t^{\mathrm{n},(j)} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n},(i)}, \boldsymbol{y}_t\right)} \tag{12}$$

2.3 Summary of the MRBPF

The MRBPF algorithm can be summarized in following steps:

(1) Initialization: take particles $\left\{\boldsymbol{x}_0^{\mathrm{n},(i)}\right\}_{i=1}^{N}$ from $p(\boldsymbol{x}_0^{\mathrm{n}})$, assign weights $\left\{w_t^{(i)}\right\}_{i=1}^{N}=\dfrac{1}{N}$, and set $\left\{\hat{\boldsymbol{x}}_{0\mid-1}^{\mathrm{l},(i)}, \boldsymbol{P}_{0\mid-1}^{\mathrm{l}}\right\}=\left\{\bar{\boldsymbol{x}}_0^{\mathrm{l}}, \bar{\boldsymbol{P}}_0^{\mathrm{l}}\right\}$, where t =1, ···, t.

(2) Evaluate and normalize the importance weights (i = 1, 2, ···, N)

$$\tilde{w}_t^{(i)}\propto\frac{p\left(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n},(i)}\right)\sum_{j=1}^{N}w_{t-1}^{(j)}p\left(\boldsymbol{x}_t^{\mathrm{n},(i)} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n},(j)}\right)}{\sum_{i=1}^{N}w_{t-1}^{(j)}q\left(x_t^{\mathrm{n},(i)} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n},(j)}, \boldsymbol{y}_t\right)}$$

$$w_t^{(i)}=\frac{\tilde{w}_t^{(i)}}{\sum_{j=1}^{N}\tilde{w}_t^{(j)}}$$

(3) Update particles and sample from the importance density

$$\left\{\boldsymbol{x}_{t+1}^{\mathrm{n},(i)}\right\}_{i=1}^{N}\sim\sum_{j=1}^{N}w_t^{(j)}q\left(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(j)}, \boldsymbol{y}_{t+1}\right)$$

(4) Kalman filter (i = 1, 2, ···, N)

$$\boldsymbol{k}_t=\boldsymbol{P}_{t\mid t-1}^{\mathrm{l}}(\boldsymbol{A}_t^{\mathrm{n}})^{\mathrm{T}}(\boldsymbol{A}_t^{\mathrm{n}}\boldsymbol{P}_{t\mid t-1}^{\mathrm{l}}(\boldsymbol{A}_t^{\mathrm{n}})^{\mathrm{T}}+\boldsymbol{B}_t^{\mathrm{n}}\boldsymbol{Q}_t^{\mathrm{n}}(\boldsymbol{B}_t^{\mathrm{n}})^{\mathrm{T}})^{-1}$$

$$\hat{\boldsymbol{x}}_{t+1\mid t}^{\mathrm{l},(i)}=\boldsymbol{A}_t^{\mathrm{l}}\left(\hat{\boldsymbol{x}}_{t\mid t-1}^{\mathrm{l},(i)}+\boldsymbol{k}_t\left(\boldsymbol{x}_{t+1}^{\mathrm{n},(i)}-f\left(\boldsymbol{x}_t^{\mathrm{n},(i)}\right)-\boldsymbol{A}_t^{\mathrm{n}}\hat{\boldsymbol{x}}_{t\mid t-1}^{\mathrm{l},(i)}\right)\right)$$

$$\boldsymbol{P}^{\mathrm{l}}_{t+1\,|\,t}= \boldsymbol{A}^{\mathrm{l}}_{t}(\boldsymbol{P}^{\mathrm{l}}_{t\,|\,t-1}-\boldsymbol{k}_t\boldsymbol{A}^{\mathrm{n}}_{t}\boldsymbol{P}^{\mathrm{l}}_{t\,|\,t-1})(\boldsymbol{A}^{\mathrm{l}}_{t})^{\mathrm{T}}+\boldsymbol{B}^{\mathrm{l}}_{t}\boldsymbol{Q}^{\mathrm{l}}_{t}(\boldsymbol{B}^{\mathrm{l}}_{t})^{\mathrm{T}}$$

From the above, it can be found that the differences between the MRBPF and RBPF[3] lie in their importance weights and particle update rules. The importance weights in the RBPF and in the MRBPF are expressed respectively by

$$w_t^{(i)} \propto \frac{p\left(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n},(i)}\right) p\left(\boldsymbol{x}_t^{\mathrm{n},(i)} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n},(i)}\right)}{q\left(\boldsymbol{x}_t^{\mathrm{n},(i)} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n},(i)}, \boldsymbol{y}_t\right)} w_{t-1}^{(i)} \tag{13}$$

and

$$w_t^{(i)} \propto \frac{p\left(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n},(i)}\right) \sum_{j=1}^{N} w_{t-1}^{(j)} p\left(\boldsymbol{x}_t^{\mathrm{n},(i)} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n},(j)}\right)}{\sum_{i=1}^{N} w_{t-1}^{(j)} q\left(\boldsymbol{x}_t^{\mathrm{n},(i)} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n},(j)}, \boldsymbol{y}_t\right)} \tag{14}$$

The particle update is achieved according to $\left\{\boldsymbol{x}_{t+1}^{\mathrm{n},(i)}\right\}_{i=1}^{N} \sim q\left(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(i)}, \boldsymbol{y}_{t+1}\right)$ and $\left\{\boldsymbol{x}_{t+1}^{\mathrm{n},(i)}\right\}_{i=1}^{N} \sim \sum_{j=1}^{N} w_t^{(j)} q\left(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(j)}, \boldsymbol{y}_{t+1}\right)$ for the RBPF and the MRBPF respectively.

2.4 Relations between the RBPF and the MRBPF

In practices, the main difference between the RBPF and the MRBPF lie in their different proposal densities defined by $q\left(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(i)}, \boldsymbol{y}_{t+1}\right)$ and $\sum_{i=1}^{N} w_t^{(i)} \cdot q\left(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(i)}, \boldsymbol{y}_{t+1}\right)$ respectively. If $q\left(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(i)}, \boldsymbol{y}_{t+1}\right) = p\left(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(i)}\right)$ is chosen, the importance weights of RBPF in Eq.(13) will be

$$w_t^{(i)} \propto p\left(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n},(i)}\right) w_{t-1}^{(i)} \tag{15}$$

and the importance weights of MRBPF in Eq.(14) will be

$$w_t^{(i)} \propto \frac{p\left(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n},(i)}\right) \sum_{j=1}^{N} w_{t-1}^{(j)} p\left(\boldsymbol{x}_t^{\mathrm{n},(i)} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n},(j)}\right)}{\sum_{j=1}^{N} w_{t-1}^{(j)} p\left(\boldsymbol{x}_t^{\mathrm{n},(j)} \mid \boldsymbol{x}_{0:t-1}^{\mathrm{n},(j)}\right)} = p\left(\boldsymbol{y}_t \mid \boldsymbol{x}_t^{\mathrm{n},(i)}\right) \tag{16}$$

The only difference between Eq.(15) and Eq.(16) is $w_{t-1}^{(i)}$, i.e., the importance weights at the moment t–1 in RBPF. Since the resampling is adopted in RBPF, $w_{t-1}^{(i)}$ should be reassigned according to $\left\{w_{t-1}^{(i)}\right\}_{i=1}^{N} = 1/N$, which, however, can be omitted after normalization. Therefore, Eq.(15) and Eq.(16) become equivalent which means the importance weights of RBPF and MRBPF are identical, and so are the importance densities in RBPF and MRBPF defined by $p\left(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(i)}\right)$ and $\sum_{i=1}^{N} w_t^{(i)} p\left(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(i)}\right)$ respectively. In fact, the resampling in RBPF can be seen as serving to produce the corresponding $w_t^{(i)}$ in $\sum_{i=1}^{N} w_t^{(i)} \cdot p\left(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(i)}\right)$, i.e., the importance density in MRBPF. As a result, despite absence of the obvious resampling in MRBPF, it is factually contained in the sampling from the importance density $\sum_{i=1}^{N} w_t^{(i)} \cdot p\left(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(i)}\right)$.

All in all, if the proposed distribution is chosen to be $q(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(i)}, \boldsymbol{y}_{t+1}) = p(\boldsymbol{x}_{t+1}^{\mathrm{n}} \mid \boldsymbol{x}_{0:t}^{\mathrm{n},(i)})$, the RBPF is equal to the MRBPF, which shows that the RBPF is nothing but a special case of the MRBPF. In common with Ref.[7], this conclusion demonstrates that the PF and the MPF will become similar when the transition prior density acts as the importance density.

3 TAN Simulations

TAN measures the variations in terrain height underneath the aircraft flight path and compares these measurements with a reference map[6] (see Fig.1). The radar altimeter provides the ground clearance (the height over the terrain), while the barometric altimeter the altitude (the height over the mean sea-level). The terrain elevation under the aircraft equals to the difference between the altitude and the ground clearance. Meanwhile, the onboard navigation computer has a digital map, which stores the terrain elevation as a function of the position. By comparing the measured terrain elevation to the digital map, the matching position in the map can be determined[6,9].

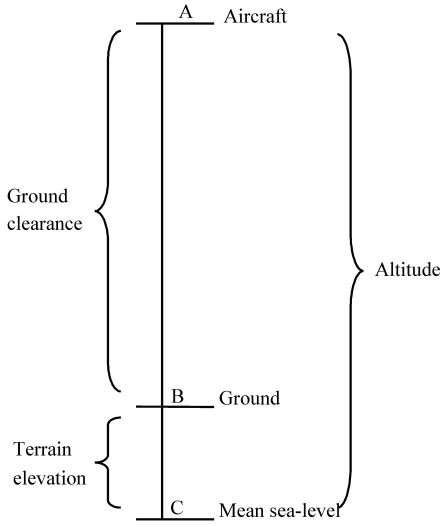

Fig.1 Principle of TAN.

3.1 Simulation model

Consider the following model [3, 9]:

$$\boldsymbol{x}_{t+1}=\begin{pmatrix}1 & T & T^2/2\\ 0 & 1 & T\\ 0 & 0 & 1\end{pmatrix}\boldsymbol{x}_t+\begin{pmatrix}T & T^3/6\\ 0 & T^2/2\\ 0 & T\end{pmatrix}\boldsymbol{\omega}_t$$

$$\boldsymbol{y}_t=h\left(50t+\boldsymbol{x}_t^1\right)+\boldsymbol{e}_t$$

where $\boldsymbol{x}_t$ is the state expressed by $\boldsymbol{x}_t=[\,\boldsymbol{x}_t^1\ \boldsymbol{x}_t^2\ \boldsymbol{x}_t^3\,]^{\mathrm{T}}$, in which $\boldsymbol{x}_t^1$, $\boldsymbol{x}_t^2$, and $\boldsymbol{x}_t^3$ define errors of position, velocity, and acceleration respectively. Both Gaussian $\boldsymbol{\omega}_t$ and $\boldsymbol{e}_t$ denote the process noise and measured noise respectively. T is the sampling period, $\boldsymbol{y}_t$ is the measurement, and h is the nonlinear function of position which corresponds to the digital map. Clearly, a closed-form expression of h does not exist because of the complexity of the terrain height variation. Fig.2 shows the correlation between the terrain elevation map and the position in the simulation. Let the constant reference speed be 50 m/s, then the TAN position simply amounts to the sum of the simulated values added by the product of reference speed and the corresponding time steps.

In order to apply the MRBPF approach, the states may be divided into two parts: linear and nonlinear, which are represented by $\boldsymbol{x}_t^{\mathrm{n}}=\boldsymbol{x}_t^1$ and $\boldsymbol{x}_t^{\mathrm{l}}=[\,\boldsymbol{x}_t^2\,,\boldsymbol{x}_t^3\,]^{\mathrm{T}}$ respectively. Then the model can be rewritten as follows

$$\boldsymbol{x}_{t+1}^{\mathrm{n}}=\boldsymbol{x}_t^{\mathrm{n}}+\begin{bmatrix}T & T^2/2\end{bmatrix}\boldsymbol{x}_t^{\mathrm{l}}+\begin{bmatrix}T & T^3/6\end{bmatrix}\boldsymbol{\omega}_t$$

$$\boldsymbol{x}_{t+1}^{\mathrm{l}}=\begin{pmatrix}1 & T\\ 0 & 1\end{pmatrix}\boldsymbol{x}_t^{\mathrm{l}}+\begin{pmatrix}0 & T^2/2\\ 0 & T\end{pmatrix}\boldsymbol{\omega}_t$$

$$\boldsymbol{y}_t=h\left(50t+\boldsymbol{x}_t^{\mathrm{n}}\right)+\boldsymbol{e}_t$$

where $T=1$, $\boldsymbol{\omega}_t\sim N\left(0,\begin{pmatrix}4 & 0\\ 0 & 0.000\,1\end{pmatrix}\right)$, $\boldsymbol{e}_t\sim N(0,36)$, $\hat{\boldsymbol{x}}_{0|1}=\begin{pmatrix}10\\ 2\\ 0.5\end{pmatrix}$, and $\boldsymbol{P}_{0|-1}=\begin{pmatrix}250\,000 & 0 & 0\\ 0 & 4 & 0\\ 0 & 0 & 0.04\end{pmatrix}$. The particle count equals to 200.

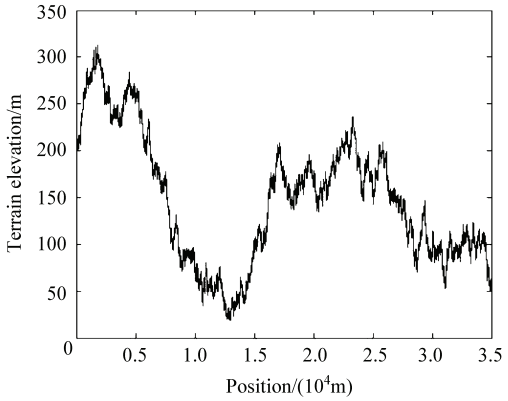

Fig.2 Terrain profile used in simulation.

3.2 Simulation results

Both RBPF and MRBPF are adopted in the simulation. All results are based on 100 Monte Carlo runs. The RBPF failed to converge in nine runs which are omitted in the aggregate statistics.

Fig.3 gives out the RMSEs of the estimated position, velocity, and acceleration errors as a function of Monte Carlo runs. The solid lines denote the results of the RBPF and the dash-dot lines the MRBPF. Note that for the position errors estimation, in comparison with the RBPF, the error variances and the RMSE of the MRBPF are decreased by 96% and 29% respectively, where the RMSE is calculated according to

$$\sqrt{\sum_{i=1}^{t}\left(\hat{\boldsymbol{x}}_i-\boldsymbol{x}_i\right)^2\Big/N}$$

where $\hat{\boldsymbol{x}}_i$ and $\boldsymbol{x}_i$ denote the estimated and true values respectively, and N is the total estimated

times. As far as the other states are concerned, since they are all estimated by the KF in both methods, no significant difference appears in the results. Table 1 shows the simulation results in detail. Fig.4 shows the numbers of the unique particles in both methods. It can be concluded that the unique particle count of the MRBPF is about 1.8 times that of the RBPF. The dependence of variances of the importance weights on Monte Carlo runs is illustrated in Fig.5, in which the variance of the MRBPF appears much lower and smoother than that of the RBPF. Furthermore, the results achieved by the MRBPF in Table 1 bears witness to its better convergence.

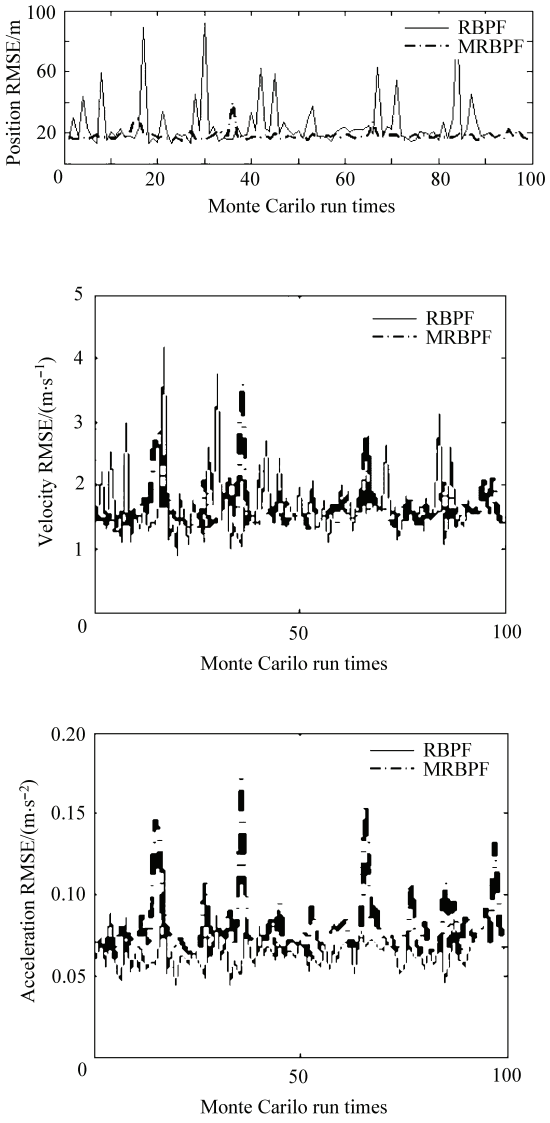

Fig.3 RMSE of state estimation versus Monte Carlo runs.

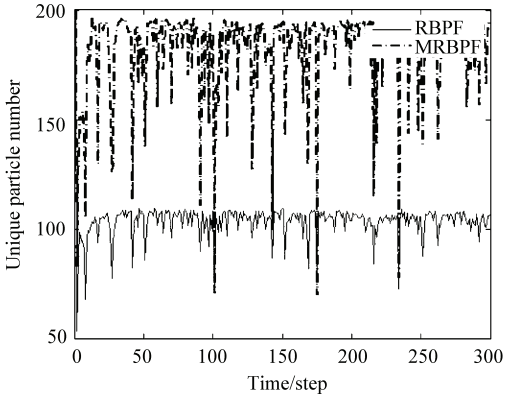

Fig.4 Unique particle numbers (200 particles used).

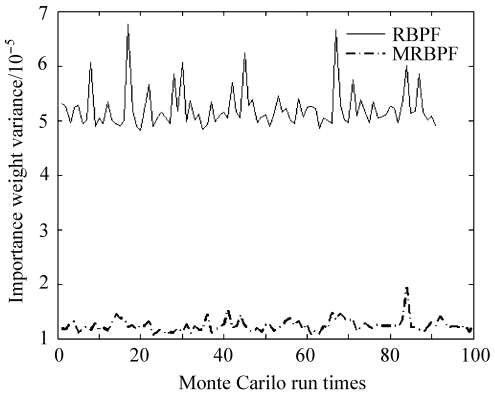

Fig.5 Variances of importance weights versus Monte Carlo runs.

Table 1 Detailed simulation results

Methods	Position error RMSE/m (variance/m^2)	Velocity error RMSE/(m·s^{-1}) (variance/(m^2·s^{-2}))	Acceleration error RMSE/(m·s^{-2}) (variance/(m^2·s^{-4}))
RBPF	25.539 023 (271.602 030)	1.700 582 (0.324 630)	0.063 854 (0.000 123)
MRBPF	18.022 465 (10.696 615)	1.619 646 (0.121 711)	0.078 043 (0.000 369)

Methods	Mean of importance weight variance	Mean number of unique particles	Divergence run times
RBPF	0.000 052	103.673 600	9
MRBPF	0.000 012	180.938 633	0

4 Conclusions

This paper presents a new filtering algorithm, called the MRBPF, for mixed linear/nonlinear state space models. It is shown that the RBPF is a special case of the proposed MRBPF. Furthermore, when applying the proposed MRBPF to TAN, the results indicate that the MRBPF gets the better of the

RBPF in terms of RMSEs of the estimated states, the estimation stability, the number of the unique particles, the variance of the importance weights, and, finally, the convergence properties of the algorithms.

References

[1] Gordon N J, Salmond D J, Smith A F M. Novel approach to nonlinear/non-Gaussian Bayesian state estimation. IEE Proceedings-F 1993; 140(2): 107-113.

[2] Steven M K. Fundamentals of statistical signal processing. Luo P F, Zhang W M, Liu Z, et al. translated. Beijing: Publishing House of Electronics Industry, 2003. [in Chinese]

[3] Nordlund P J. Recursive state estimation of nonlinear systems with applications to integrated navigation. Linkoping: Linkoping University, LiTH-ISY-R-2321, 2000.

[4] Doucet A, Freitas N, Gordon N J. Sequential Monte Carlo in practice. New York: Springer Press, 2001.

[5] Pitt M K, Shephard N. Filtering via simulation: auxiliary particle filters. JASA 1999; 94(446): 590-599.

[6] Bergman N. Recursive Bayesian estimation navigation and tracking applications. PhD thesis, Linkoping University, 1999.

[7] Klaas M, Freitas N, Doucet A. Toward practical N^2 Monte Carlo: the marginal particle filter. Proceedings of the 21st Annual Conference on Uncertainty in Artificial Intelligence (UAI-05). Arlington, Virginia: AUAI Press, 2005; 308-315.

[8] Klaas M. Exorcising N^2 stigmata in sequential Monte Carlo. MS thesis, University of British Columbia, 2005.

[9] Nordlund P J. Sequential Monte Carlo filters and integrated navigation. PhD thesis, Linkoping University, 2002.

[10] Schon T, Gustafsson F, Nordlund P J. Marginalized particle filters for mixed linear/nonlinear state-space models. IEEE Transaction on Signal Processing 2005; 53(7): 2279-2289.

[11] Schon T, Karlsson R, Gustafsson F. The marginalized particle filter in practice. Linkoping: Linkoping University, LiTH-ISY-R-2715, 2005.

Biographies:

Yin Jianjun Born in 1980, he received a B.S. degree from the Department of Electronic Engineering, Fudan University in 2003. Now he is working towards a doctorate degree in Fudan University. He takes an academic interest in sequential Monte Carlo methods, navigation and tracking.

E-mail: yinjianjun@fudan.edu.cn

Zhang Jianqiu Born in 1962, he received an M.S. and a Ph.D. degree from the Department of Electrical Engineering, Harbin Institute of Technology. Now he is a professor in the Department of Electronic Engineering, Fudan University, Shanghai, China, and an IEEE senior member. His main academic interest focuses on digital signal processing and application in advanced sensors, intelligent instrumentation systems and control and communications.

E-mail: jqzhang01@fudan.edu.cn

Mike Klaas Born in 1981, he graduated with a M.S. in Computer Science from the University of British Columbia, Canada, and now he is working for UBC's Laboratory of Computational Intelligence. His research interests include generalized Monte Carlo methods and fast computational techniques for statistical inference.

E-mail: klaas@cs.ubc.ca